UCS645 – Parallel & Distributed Computing

Name: Lavish Arora
Rolll no: 102483041
Group no:3C43

Assignment 2: Performance Measurement using OpenMP

Question 1: Molecular Dynamics Force Calculation

Aim:

To study the effect of OpenMP parallelization on molecular dynamics force calculations.

Problem

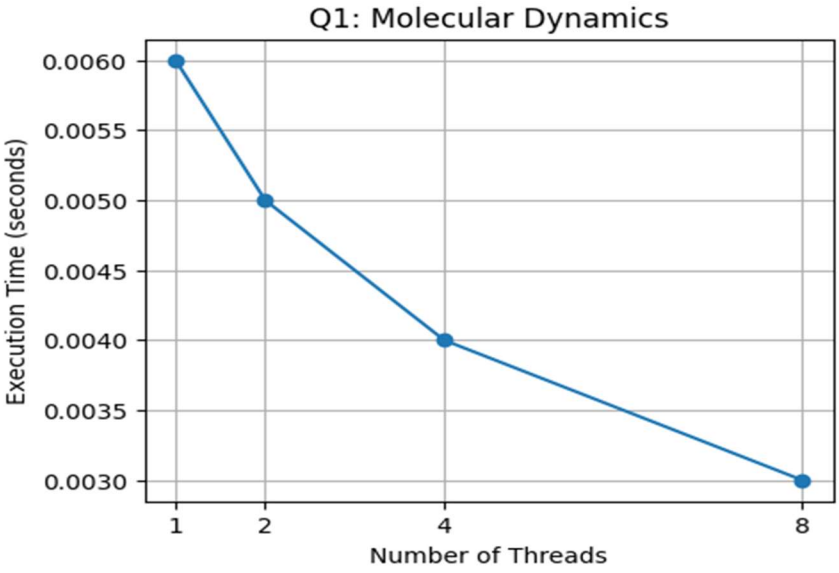
Molecular dynamics simulations require computing pairwise forces between particles, which results in high computational complexity. As the number of particles increases, execution time grows rapidly, making optimization essential.

Methodology:

The outer loop of force computation is parallelized using OpenMP to distribute particle interactions across multiple threads. Atomic operations are used to safely update shared force values, and execution time is measured.

Results:

Number of Threads	Execution Time (seconds)
1	0.006
2	0.005
4	0.004
8	0.003



Terminal Snapshort:

```
PS C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> $env:OMP_NUM_THREADS=1; .\q1
Execution Time: 0.006000 seconds
PS C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> $env:OMP_NUM_THREADS=2; .\q1
Execution Time: 0.005000 seconds
PS C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> $env:OMP_NUM_THREADS=4; .\q1
Execution Time: 0.004000 seconds
PS C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> $env:OMP_NUM_THREADS=8; .\q1
Execution Time: 0.003000 seconds
PS C:\Users\asus\OneDrive\Desktop\UCS645\LAB2>
```

Question 2: DNA Sequence Alignment (Smith–Waterman)

Aim:

To analyze the performance of DNA sequence alignment using OpenMP.

Problem

DNA sequence alignment uses dynamic programming to compute similarity scores between sequences. Each cell in the scoring matrix depends on previously computed neighboring cells, creating strong data dependencies.

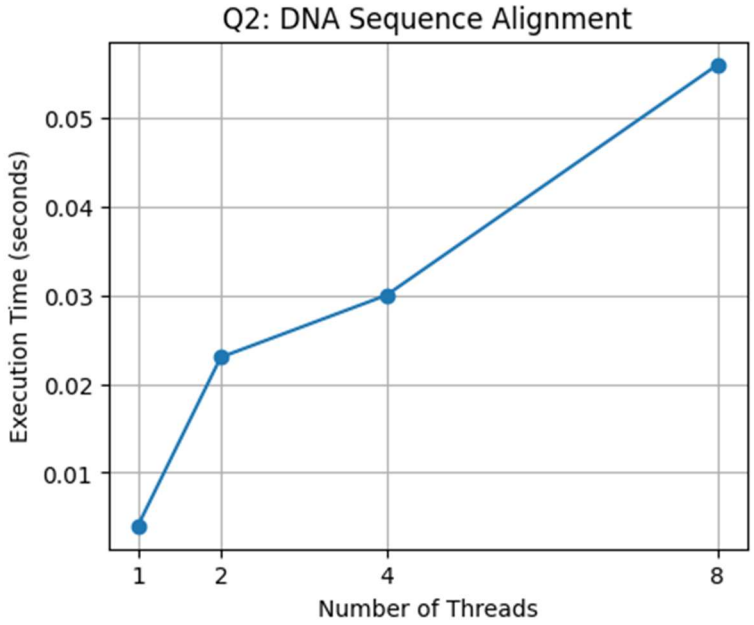
Methodology

Wavefront parallelization is implemented using OpenMP, where independent diagonals of the matrix are computed in parallel. Due to synchronization and dependency constraints, additional overhead is introduced during parallel execution.

Wavefront parallelization is applied along matrix diagonals.

Results:

Number of Threads	Execution Time (seconds)
1	0.004
2	0.023
4	0.03
8	0.056



Terminal Snapshort:

```
S C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> gcc -fopenmp q2.c -o q2
S C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> $env:OMP_NUM_THREADS=1; .\q2
xecution Time: 0.004000 seconds
S C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> $env:OMP_NUM_THREADS=2; .\q2
xecution Time: 0.023000 seconds
S C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> $env:OMP_NUM_THREADS=4; .\q2
xecution Time: 0.030000 seconds
S C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> $env:OMP_NUM_THREADS=8; .\q2
xecution Time: 0.056000 seconds
S C:\Users\asus\OneDrive\Desktop\UCS645\LAB2>
```

Question 3: Heat Diffusion Simulation

Aim:

To evaluate the scalability of heat diffusion simulation using OpenMP.

Problem

Heat diffusion simulation models the transfer of heat across a 2D grid, where each cell's temperature depends on its neighboring cells. This type of computation is commonly used in scientific simulations.

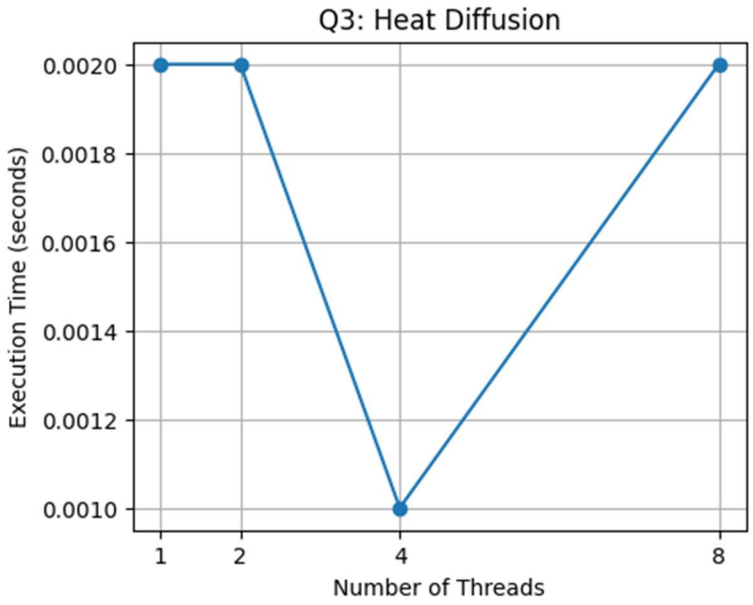
Methodology

The nested loops updating the grid are parallelized using OpenMP's `collapse` directive, allowing multiple grid points to be processed simultaneously. Execution time is measured to analyze scalability with increasing thread counts.

.

Results:

Number of Threads	Execution Time (seconds)
1	0.002
2	0.002
4	0.001
8	0.002



Terminal Snapshot:

```
PS C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> notepad q3.c
PS C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> gcc -fopenmp q3.c -o q3
PS C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> $env:OMP_NUM_THREADS=1; .\q3
Execution Time: 0.002000 seconds
PS C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> $env:OMP_NUM_THREADS=2; .\q3
Execution Time: 0.002000 seconds
PS C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> $env:OMP_NUM_THREADS=4; .\q3
Execution Time: 0.001000 seconds
PS C:\Users\asus\OneDrive\Desktop\UCS645\LAB2> $env:OMP_NUM_THREADS=8; .\q3
Execution Time: 0.002000 seconds
PS C:\Users\asus\OneDrive\Desktop\UCS645\LAB2>
```